

553,577

## (12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织  
国际局

(43) 国际公布日:

2004年10月28日(28.10.2004)

PCT

(10) 国际公布号:

WO 2004/092946 A1

- (51) 国际分类号<sup>7</sup>: G06F 7/00
- (21) 国际申请号: PCT/CN2004/000375
- (22) 国际申请日: 2004年4月19日(19.04.2004)
- (25) 申请语言: 中文
- (26) 公布语言: 中文
- (30) 优先权:  
03122702.3 2003年4月17日(17.04.2003) CN  
03143358.8 2003年9月27日(27.09.2003) CN
- (71)(72) 发明人/申请人: 李志中(LI, Zhitzhong) [CN/CN];  
中国浙江省武义县东升东路南七巷11号, Zhejiang  
321200 (CN)。
- (72) 发明人: 及
- (75) 发明人/申请人(仅对美国): 徐菊园(XU, Juyuan) [CN/];  
中国浙江省武义县东升东路南七巷11号,  
Zhejiang 321200 (CN)。
- (74) 代理人: 中国专利代理(香港)有限公司(CHINA  
PATENT AGENT (H.K.) LTD.); 中国香港湾仔港湾  
道23号鹰君中心22字楼, Wanchai, Hong Kong (CN)。

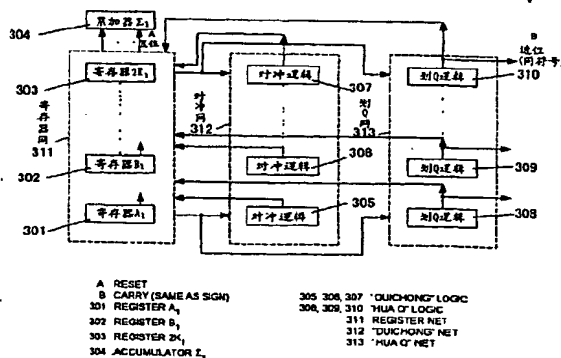
(81) 指定国(除另有指明, 要求每一种可提供的国家保护):  
AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW,  
BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM,  
DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR,  
HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC,  
LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN,  
MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL,  
PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM,  
TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA,  
ZM, ZW

(84) 指定国(除另有指明, 要求每一种可提供的地区保护):  
ARIPO(BW, GH, GM, KE, LS, MW, MZ, SD, SL, SZ,  
TZ, UG, ZM, ZW), 欧亚专利(AM, AZ, BY, KG, KZ,  
MD, RU, TJ, TM), 欧洲专利(AT, BE, BG, CH, CY,  
CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT,  
LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI(BF,  
BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR,  
NE, SN, TD, TG)

本国际公布:  
— 包括国际检索报告。

所引用双字母代码和其它缩写符号, 请参考刊登在每期  
PCT公报期刊起始的“代码及缩写符号简要说明”。

(54) Title: DIGITAL ENGINEERING METHOD AND PROCESSOR OF THE MIXED Q N-ARY AND CARRY LINE  
(54) 发明名称: 混Q进制、进位行数字工程方法和处理器



(57) Abstract: The present invention relates to the field of digital engineering method and processor, it provides a new digital engineering method, improves operational speed. The mixed Q N-ary and carry line of digital engineering method according to the present invention includes: A digital sign is added to every bit of the common Q N-ary digitals participating in the operation, there are k mixed N-ary digitals to participate in the operation. Sums up k mixed Q N-ary at the same time. Adding by bit beginning with the lowest bit, i. e., at a certain bit, two digitals of the above described k digitals are taken and added to generate "addition by bit", and the sum is taken into the next operation layer as "partial sum", meanwhile the acquired "mixed digital carry" is put into the high-order bit which is close to the bit that is any carry line of the next operation layer. The operations don't stop until it generates "mixed Q N-ary carry" line. Then the sum obtained by the last "addition by bit" is the result if addition operation. The present provides a processor with mixed Q N-ary and carry line operation.

[见续页]

WO 2004/092946 A1



## (57) 摘要

本发明涉及数字工程方法和处理器领域，提出一种新的数字工程方法，提高运算速度。本发明的混 Q 进制、进位行数字工程方法包括：

将参与运算的普通 Q 进制数的每一位数字都加上一个数符，参与运算的数为 k 个混 Q 进制数，对 k 个数同时进行混 Q 进制的求和运算，从最低位开始，按位相加，即在某一位上，取前述 k 个数中的二个按位相加，得到“按位和”，将此和数记入下一运算层，作为“部份和”数，同时所得“混数进位”，则存放到下一运算层中的任一进位行中与该位相邻的高位处，经过对此运算，直至不产生“混 Q 进位”为止，则最后一次“按位加”所得和数，即为所求加法运算结果。本发明提供了混 Q 进制、进位行处理器。

## 混 Q 进制、进位行数字工程方法和处理器

## 技术领域

本发明涉及数字工程方法和处理器领域，特别是处理器的运算器

## 背景技术

四则运算是数的最基本运算。正如恩格斯所说：“四则（一切数学的要素）”<sub>(1)</sub>，加法又是四则运算的最基本的运算。因此，我们理所当然应当对四则运算，尤其是对加法运算给予特别的关注。当前电子计算机中数学的四则运算，首先是加法，有许多不尽如人意之处。主要表现为运算速度慢；在减法中，未能充分利用负数的作用，而且，不能“连减”。尤其在加减混合运算中，不能一步到位；在乘法中，加法的缺点更加扩大严重；在除法中，上述缺点依旧。总之，在最小的数体——有理数体中，四则运算情况并不满意。

在数字工程方法中，特别是在中小学数学教科书中有大量的数值运算。对运算的解剖，表明存在一些隐含的操作程序，以至产生“隐患”。以加法为例。例一“两数相加”。算式如式一。其中，十位上的和数 3，解剖一下，其微程序操作是：

$$123456+345678=469134$$

$$78+297+259=634$$

$$\begin{array}{r} 123456 \\ + 345678 \\ \hline 469134 \end{array}$$

$$\begin{array}{r} 78 \\ 297 \\ + 259 \\ \hline 634 \end{array}$$

式一

式二

④个位上来的进位（见标志）

⑤十位上 5、7 两数字与低位进位相加，即（5+7+1）。取其和的个位。

⑥上列（5+7+1）和的进位送到高位（见标志）。其余各位情况类似。

又如，例二，设三数求和，算式如式二：

$$78+297+259=634$$

如图可见，上述情况更为加重。

显然，存在下列缺点：

a、进位标示困难。若用小数字表明，则易混淆且字面积受限。特别是表 456789 时就更烦人；若以“.”字写在数字间，则易与小数点混淆且表示 456789 也不便；若以手指数数，则速度慢且不方便；若心算，则费脑力且易错。总之，比较讨厌，易出错。

b、一般两数相加时，每一位上要有三个数相加求和。于是，需二次运算。三及三以上个数求和时，则更不方便。

c、验算困难。一般采用重做一遍，费时费力。

②减法比加法麻烦。且不能在同一竖式中“连减”，必须断开。特别在加减混合运算时，不能一步到位。

③乘法中，这类情况更为严重。而且，加减乘除运算格式不统一，除法时另起炉灶。

另一方面，在电子计算机方式数字工程中，同样有大量的数值运算。这些数一般均采用普通二进制数制{2}来表示。其负数常以原码、反码、补码、移码之类来表示。在现有计算机中运算均以二个数运算，而无法实现“多重运算”。所谓“多重运算”是指多于二个数同时进行加减。

在采用其他{Q}等普通数制的电子计算机中，存在相应的许多复杂性。

## 发明内容

本发明提出一种新的数字工程方法，提高运算速度，同时加强运算正确性的保障，使出错的可能性显著减少。

本发明的另一个目的是提供一种新的处理器，在现有制造技术的基础上，在设备量相近的情况下，大大提高电子计算机的运算速度。

根据本发明的一个方面，提供一种混 Q 进制、进位行数字工程方法，包括以下步骤：

第 1 步，将参与运算的普通 Q 进制数的每一位数字都加上一个数符，即表示该位数为正或负，使它成为混 Q 进制数，设，参予运算的数为 k 个混 Q 进制数，

第 2 步, 对  $k$  个数同时进行混  $Q$  进制的求和运算, 从最低位开始, 按位相加, 即在某一位上, 取前述  $k$  个数中的二个按位相加, 得到“按位和”, 即为该位这二个按位相加的和数, 将此和数记入下一运算层, 作为“部份和”数, 同时所得“混数进位”, 则存放到下一运算层中的任一进位行中与该位相邻的高位处;

第 3 步, 在该位上取  $k$  个数中的另二个按位相加, 进行第二步的运算, 如此反复, 直至  $k$  个数均取完为止; 当  $k$  个数中仅剩下一个数时, 则直接移至下一运算层的同一位上作为“部份和”数;

第 4 步, 在上述某位的相邻高位上, 重复第 2 步及第 3 步的运算, 直至  $k$  个运算数的每一位都全部操作结束;

第 5 步, 在下一个运算层中, 将上述“按位和”数与进位行中的“进位数”进行前述第 2 步、第 3 步、第 4 步求和运算;

重复第 2 步至第 5 步的运算, 直至不产生“混  $Q$  进位”为止, 则最后一次“按位加”所得和数, 即为所求加法运算结果。

根据本发明的另一个方面, 提供一种混  $Q$  进制、进位行处理器, 包括: 输入逻辑,  $k$  重运算器, 输出转换逻辑及控制器组成; 混  $Q$  进制数移位寄存器输入逻辑至  $k$  重运算器;  $k$  重运算器中, 混  $Q$  进制数经  $k$  重运算获得混  $Q$  进制数的结果, 经由译码器输出转换逻辑以  $Q$  进制数或混  $Q$  进制数、或普通十进制数通过输出逻辑输出, 控制器协调控制整个运算控制器的逻辑; 其中,

为  $2K$  个寄存器中的每个寄存器的每一位分配一个符号位, 该符号位为普通二态触发器; 前面  $K$  个寄存器存放输入的  $K$  个混  $Q$  数, 后面  $K$  个寄存器组成  $K$  个进位行;

在运算过程中, 其中两个寄存器的某位, 经累加器累加得到该位的和与高位进位, 其中进位送至任一进位寄存器的相邻高位; 当下一个运算命令到达时, 将进位行与原存数送入累加器相加;

如此重复, 最后再经累加器获得所求和数。

附图说明 (以混二进制为例)

图 1 是混  $Q$  进制计算机总逻辑框图。

图 2 是运算控制逻辑框图;

图 3 是  $K$  重运算器一位的逻辑框图;

图 4 是对冲逻辑 (对冲器) 的逻辑框图;

图 5 是划 Q 逻辑 (划 Q 器) 的逻辑框图;  
具体实施方式

## 1、《进位行方法》

### 1.1 进位与《进位行方法》

在电子计算机中, 运算速度提高的关键之一, 就在于“进位”。进位的获得, 进位的存贮以及进位的参予运算都是至关重要的。“进位”就是争“速度”。在笔算中, 还直接影响到“出错率”。

所谓《进位行方法》就是, 在运算过程中, 将产生的进位存放在参予运算的位置, 然后直接进行运算的方法。通常, 将同运算层各位上的进位排列成一行, 称为“进位行”。(运算层的概念, 见下节)

举例如下, 设两普通十进制数求和, 算式以竖式求和。如式三:

$$\begin{array}{r}
 123456+345678=469134 \\
 \hline
 345678 \\
 468024 \dots\dots \oplus \text{ 行} \\
 111 \dots\dots \text{进位行} \\
 \hline
 469134
 \end{array}$$

式三

为简化起见, 这里将横竖式合写。个位运算  $(6+8)=14$ , 其进位 1 写于下一行的高一位上。依此类推。

式中二数相加时, 各位上不计进位的求和, 称为“按位加”。其和称为“按位和”。按位和的运算行, 称为“ $\oplus$ 行”。

各进位排成的行, 称为“进位行”。由  $\oplus$  行与进位行组成“运算层”。

式中一些“+”号已省去。以后可以知道, 在《混进方法 HJF》中, 各个“运算层”只存在一种运算, 这就是“+”。故可以不必在运算层中写出“+”号。

### 1.2 《进位行方法》分析

#### 1.2.1 二数求和的分析

采用《进位行方法》的加法运算由上节可知:

① 两数相加时, 每一位上只有二个数字相加, 不可能二个以上数相加;

② 在进位行中直接标示进位, 不存在任何困难;

### ③ 验算十分方便。

[引理一] 两数相加时，任意位上要么有进位记为 1，要么无进位记为 0；

[引理二] 两数相加时，任意位上的 $\oplus$ 和可为 0~9 之一。但是，当该位上有向高位进位时，该位上的 $\oplus$ 和只能为 0~8 之一，而不能为 9。

由[引理一]和[引理二]可得：

[定理一] 两数相加时，当且仅当某位上没有向高位进位时，该位上的 $\oplus$ 和才可能出现 9。

#### 1.2.2 层次概念及运算层

设两数求和。算式为式四、式五

5843029 + 4746979 = 10590008	
$  \begin{array}{r}  5843029 \\  4746979 \\  \hline  9589998 \\  11\ 1111 \\  \hline  10590008  \end{array}  $	} 运算层
$  \begin{array}{r}  5843029 \\  + 4746979 \\  \hline  9589998 \\  1\quad 1 \\  \hline  0589908 \\  1\quad 1 \\  \hline  10589008 \\  1 \\  \hline  10580008 \\  1 \\  \hline  10590008  \end{array}  $	} 第一运算层 } 第二运算层 } 第三运算层 } 第四运算层

式五

式四

由式四可见，运算是分层次进行的，每一运算层，仅完成一项简单运算。

这就是运算的“层次”概念，运算层将一个运算解剖成微运算、子运算。

“层次”概念在数学中是基本概念。《进位行方法》正是建立在此概念基础上。以往的加法运算方法，本质上也隐含“层次”概念。因此，《进位行方法》中的“层次”从总体上看，并未增加运算的复杂性。反之，以往的方法由于隐含了“层次”，反而进一步增加了运算的复杂性。这一点，也进一步造成运算速度被明显降低。两者对比，就会一清二楚。

在《进位行方法》中，两数相加的各个运算层，可以合并为一个运算层。如式五，请见进一步分析。

### 1.2.3 唯一的运算层

两数相加时，特别情况下会出现多次运算层。各层有如下关系成立。

[引理三] 二数相加，当某位前一运算层上有进位时，其后各运算层上均不可能出现进位。（由引理一、二得）

[引理四] 二数相加，当某位后一运算层上有进位时，其前各运算层上必无进位。（由引理一、二得）

[定理二] 二数相加时，同一位各运算层上，要么都无进位，要么只能有一个进位。（由引理三、四得）

[推论] 可以将全部各层进位行合并为一个进位行，各运算层合并为一个运算层。（也可将非第一运算层的进位以小圆圈标示，见式五）

$  \begin{array}{r}  231 \\  786 \\  +989 \\  \hline  8\cancel{6}6 \\  1\cancel{1}\cancel{1} \\  \textcircled{1}\textcircled{1}\cancel{3} \\  \hline  2006  \end{array}  $	$  \begin{array}{r}  786 \\  \cancel{6}6\cancel{6} \\  57\cancel{5} \\  32\cancel{1} \\  699 \\  + \quad 999 \\  \hline  1\cancel{6}16 \\  223 \\  \textcircled{1}2 \\  \hline  4046  \end{array}  $
--	--

式六

式七

### 1.2.4 三数及三数以上求和分析

设三数求和，算式为  $231+786+989=2006$  （见式六）

操作要点：

#### ① “划十”的运用；

所谓“划Q”，即Q进位的两数在某位上相加时，其按位加和为零，但该位上产生进位（与两数符号一致）。进位放入进位行；同时，在某位上，该两数均不再参加运算。

在十进制时即为“划十”，具体说明如下：

a、同一位上两数和为“十”时，可在算式中将两数字以斜线划



去, 然后在高位上补 1。

b、同一位上几数和为 20、30、40... 等时, 可将几数字均划去, 然后在高位上补 2、3、4... 等。

又, 设六数求和。算式为  $786+666+575+321+699+999=2046$  (见式七)。

②多个数相加, 会出现二个及二个以上的运算层。为了减少运算层数, 同一位上的同一运算层空位中, 进位及  $\oplus$  位数可以任意占位。

③尽量减少运算层。

a、较小的数, 直接合并算;

b、尽量在“配对”中进位;

c、尽量减少在第一运算层上相加数的个数, 尽量使第二及二以上运算层不出现。

④同一位上, “相同数”、“连续数”等可直接获得“部分和”。

⑤设有  $m$  个数求和。( $m$  为  $\geq 2$  的自然数。) 总运算层以  $n$  来表示。( $n$  为非负整数)。则:

$$\begin{cases} n_{\min}=0 \text{ (通常 } n=0, 1, 2, \text{ 而以 } n=1 \text{ 为最常见)} \\ n_{\max}=\begin{cases} m/2, & m \text{ 为偶数时} \\ m+1/2, & m \text{ 为奇数时} \end{cases} \end{cases}$$

式八

## 2、混数及混数数制

### 2. 1 《数制理论》

2.1.1 按同一种规则记录数, 便于用来在一个数系统中进行运算的数的制度, 称为“记数系统的制度”。简称为“数制”。一个数的质, 首先就是由其所属的数制来决定的。恩格斯指出: “单个的数在记数法中已经得到了某种质, 而且质是依照这种记数法来决定的。”

“一切数的定律都取决于所采用的记数法, 而且被这个记数法所决定。”  $\langle - \rangle$

《数制理论》就是研究数制的生成、分类、分析、比较、变换等以及数在各邻近学科与实践应用的科学。它是数学的基础理论之一。

数制是数的属性。不存在没有所属数制的数, 也不存在没有所属

数的数制。[文中凡未标明数制的数，均指普通十进制数。下同。]

### 2.1.2 位值制数制

设，构造一个数系的数由各不相同位置上的“数符”来表示。“数符”又称“数字”，通常从右向左水平排列，其相应的数值由低（小）到高（大）。每个数位上的数字给定一个单位值（又称“位值”），由此来表示整个数系中每一个数的数制不变，称为“位值制数制”。

我们以下讨论的数制，都是“位值制数制”。简称为“数制”。所讨论的数均约定为整数。

### 2.1.3 数制有三大要素：数位 $I$ ，数元集 $Z_i$ 和权 $L_i$ 。

a、数位  $I$ ，表示数制中数的各位数字的位置。以  $I$ （序数）从右自左来表示。即， $i=1, 2, 3, \dots$  表示该数的第 1, 2, 3,  $\dots$  位。

b、数元集  $Z_i$ ，表示第  $I$  位上的“数元”组成的集合。同一数制系统中，各个数同一位上不同符号的全体，组成一个该位上的数符集。该数符集中的元素，称为“数的元素”。简称为“数元”。因此，该数符集称为“数元集”。

数元集  $Z_i$  可以随着  $i$  的取值不同而不同，也可以相同。

数元集  $Z_i$  中的数元可为复数或其他多种多样符号。以  $a_j$  来表示数元（ $a_1, a_2, a_3, \dots$ ）以  $ia_j$  表示第  $i$  位上数元  $a_j$ （ $j$  为自然数）

数元集  $Z_i$  的基数  $P_i$ （ $P_i$  为  $\geq 2$  的自然数）表示了集的元素总数。它“不但决定它自己的质，而且也决定其他一切数的质。” $\langle -, \rangle P_i$  的取值不同，标示了数元集  $Z_i$  的变化。各位上的  $P_i$  均相同，则称为“单一基数”；否则，称为“混合基数”。相应的数制，称为“单一数制”及“混合数制”。

c、权  $L_i$ ，表示第  $i$  位上的位值大小。特称此位值为“权  $L_i$ 。”

$L_i$  为实数（由于复数集非有序体，故不采用），不同的  $L_i$ ，就决定了不同的位值。

在“编码理论”中，“编码”的主要特征就在于权  $L_i$ 。

实际中常见的权  $L_i$  采用所谓“幂权”。即，令  $L_i = Q^{(i-1)}$ ， $Q_i$  为实数。为便于计算起见，常取  $Q_i$  为自然数。常见各位  $L_i$  均为幂权，而且成等比  $Q$  的数制。 $Q$  称为数制幂权的“底数”或数制的“底数”。底数  $Q$  的不同，决定了不同的  $L_i$ ，从而决定了不同的数值。通常，称这种数制为“ $Q$  进制  $\{Q\}$ ”。

另一种常用的权  $L_i$  采用“等权”，即各位上的权相同。

## 2.2 混数及混数数制

根据上述数制的三大要素，数制可以有无穷无尽的种类。

当数元集  $Z_i$  中，基数  $P_i$  各位均相同时， $P_i = P_{i+1} = P$  称为“单一基数”；各位上  $P_i$  不同时，称为“混合基数”。与此相应的数制称为“单一数制”和“混合数制”。

当  $Q=2, 3, 10$  等时，相应的数制就被称为“二进制”、“三进制”、“十进制”等。

一个数制：当  $p=Q$  时，自然数在该数制可以连续唯一的形态表达，称为“连续数制”，又称“普通数制”；

当  $P > Q$  时，自然数在该数制可以连续，但有时以多种形态表达，称为“重复数制”；

当  $P < Q$  时，自然数在该数制只能断续的形态表达，称为“断续数制”。

当数元集  $Z_i$  中，含数元 0 时，该相应数制被称为“含 0 数制”；

当数元集  $Z_i$  中，全部数元为连续整数时，该相应数制被称为“整数段数制”；

当数元集  $Z_i$  中，既有正数元，又有负数元时，相应数制被称为“混数数制”；混数数制中的数，称为“混数”。“混数”中既有正数元又有负数元的数，称“纯混数”。在  $\{Q^*\}$  数中，既有正数元又有负数元的数，称为“纯  $\{Q^*\}$  数”。（ $\{Q^*\}$  定义见下一节。）

当数元集  $Z_i$  中，正负数元是相反数时，相应数制称为“对称数制”；显然，“对称数制”是“混数数制”的一种。

## 2.3 混 $Q$ 进制 $\{Q^*\}$ 和普通混 $Q$ 进制 $\{\text{普 } Q^*\}$

在《数制理论》中，一个数制的名称采用“ $Z_i L_i$ ”。例如  $\{0, 1, 2, \}$  三进制；或者  $Z_i$  以文字表明其特征。

对于普通十进制，在《数制理论》中，它的名称是：

“单一基数  $P=10$  的，含 0 的，整数段，非负不对称的十进制”。可写为  $\{+, \text{含 } 0, \text{整数段, 非负}\}$  十进制，或者写为  $\{0, 1, 2, \dots, 9\}$  十进制。一般情况下，我们进一步缩写为  $\{+\}$ ，称为“普通十进制”。

对于普通二进制在《数制理论》中，它的名称是：

“单一基数  $P=2$ ，含 0，整数段，非负不对称的二进制”。可写为  $\{=, \text{含 } 0, \text{整数段, 非负}\}$  二进制，或者写为  $\{0, 1\}$  二进制。一般情况下，我们进一步缩写为  $\{=\}$ ，称为“普通二进制”。

本文中《混数、进位行方法》（简称《混进方法 HJF》见下一节。）中的混数数制主要有三类。在《数制理论》中，它们的名称分别是：

“单一基数  $P=19$ ，含 0，整数段，对称的十进制”。可写为  $\{十九, \text{含 } 0, \text{整数段, 对称}\}$  十进制，或者写为  $\{0, \pm 1, \pm 2, \dots, \pm 9\}$  十进制。一般情况下，我们进一步缩写为  $\{+*\}$ ，称为《混十进制》（用于笔算方式数字工程，特别是有理数运算教科书等时）。或者，

“单一基数  $P=3$ ，含 0，整数段，对称的二进制”。可写为  $\{三, \text{含 } 0, \text{整数段, 对称}\}$  二进制，或者写为  $\{0, \pm 1\}$  二进制。一般情况下，我们进一步缩写为  $\{=*\}$ ，称为《混二进制》（用于计算机等时）。同样，对于  $\{0, \pm 1, \dots, \pm (Q-1)\}$   $Q$  进制缩写为  $\{Q*\}$ ，称为《混  $Q$  进制》。

在混数数制中，另一类为普通数制“ $Q$ ，含 0 整数段，对称  $Q$  进制”，称为“含 0，整数段，对称，普通  $Q$  进制”，又称为“普通混  $Q$  进制”  $\{\text{普 } Q*\}$ 。其中典型的是  $\{\bar{1}, 0, 1\}$  三进制，称为“普通混三进制”  $\{\text{普三}*\}$ 。[注：令负  $A$  表为  $\bar{A}$ ，读作负  $A$ 。如，负  $1=\bar{1}$ 。下同。]显然，普通混  $Q$  进制中  $Q$  只能为  $>1$  的奇数。

在不含 0 的混数数制中，有一类为普通数制“ $Q$ ，不含 0，整数段，对称  $Q$  进制”，称为“不含 0，整数段，对称，普通  $Q$  进制”，又称为“不含 0 普通混  $Q$  进制”  $\{\text{不含 } 0 \text{ 普 } Q*\}$ 。其中典型的是  $\{\bar{1}, 1\}$  二进制，称为“不含 0 普通混二进制”  $\{\text{不含 } 0 \text{ 普二}*\}$ 。显然，不含 0 普通混  $Q$  进制中， $Q$  只能为正偶数。

除上述三类“对称混数数制”外，其他对称混数数制，称为“其他对称混数数制”；其他不对称混数数制，称为“非对称混数数制”。

### 3、《混进方法 HJF》及其混十进制 $\{+*\}$ 四则运算。

采用混数和《进位行方法》来进行有理数运算的方法，称为《混数、进位行方法》，简称为《混进方法 HJF》。当用于笔算方式数字工程，特别是有理数运算教科书等之中时，采用的是  $\{+*\}$  混十进制的《混进方法 HJF》。当用于电子计算机等之中时，采用的是  $\{=*\}$  混二进制等的《混进方法 HJF》。

## 3. 1 { + \* } 的加法

例  $1\bar{2}3 + 4\bar{5}\bar{6} = 427$  如式九

$$\begin{array}{r}
 1\bar{2}3 \\
 + 4\bar{5}\bar{6} \\
 \hline
 5\bar{7}\bar{3} \\
 \hline
 5\ 2\ 6 \\
 \bar{1}\ 1 \\
 \hline
 4\ 2\ 7
 \end{array}$$

## 式九

式中求得和为  $5\bar{7}\bar{3}$ 。当需要转化为普通十进制 {+} 数时, 和为 427。

一般来说, 所求和  $5\bar{7}\bar{3}$  不必转化 (特别是作为计算过程中间结果时)。确需转化时, 方法见 4.1 转换法则。

## 3. 2 { + \* } 的减法

3. 2. 1 例  $1\bar{2}3 - 4\bar{5}\bar{6} = 1\bar{2}3 + \bar{4}\bar{5}\bar{6} = \bar{3}39$ 

首先化为加法来运算, 这是由于混数的特性所决定。这一来, 实际计算中, 加减就合并为加法了。这就消除了通常连加减的困难。

例  $112 + 56 - 32 - 85 + 67 - 46 = 72$ 

$$\begin{array}{r}
 2\bar{3}\bar{8} \\
 \times 8\bar{9} \\
 \hline
 \bar{1}\bar{8}\bar{7}2 \\
 \bar{1}2\bar{7} \\
 6\bar{4}\bar{4} \\
 \hline
 1\bar{2}\bar{6} \\
 12102 \\
 \bar{6} \\
 \hline
 12\bar{5}02
 \end{array}
 \qquad
 \begin{array}{r}
 1\bar{4}2 \\
 5\bar{6} \\
 \bar{3}\bar{2} \\
 \bar{8}\bar{5} \\
 67 \\
 + \bar{1}\bar{4}\bar{6} \\
 \hline
 52 \\
 2 \\
 \hline
 72
 \end{array}$$

## 式十

## 式十一

3. 2. 2 约混。这是指二数求和时, 同一位上的相反数可以消去。也可称为“对消”或“对冲”。在算式中, 可以斜线划去。也就是说, 所谓“对冲”, 即两相反数, 其和为零。该某位上的两数不再参加以后的运算。

## 3. 3 { + \* } 的乘法

例  $2\bar{3}8 \times 8\bar{9} = 12\bar{5}02$

### 3.4 {+\*} 的除法

例  $5728 \div 23 = 249 \dots 1$

要点：① 式一采用原普通除法，现采用四则统一算式如式十三

② 式中  $57 - 23 \times 2 = 57 + \bar{23} \times 2 = 57 + \bar{46}$  也就是说，由于采用混数可使除法中的“减”过程变为“加”的过程。其余同此。

$$\begin{array}{r}
 23 \overline{) 5728} \\
 \underline{46} \phantom{00} \\
 112 \\
 \underline{92} \phantom{00} \\
 208 \\
 \underline{207} \phantom{00} \\
 1
 \end{array}$$

$$\begin{array}{r}
 5728 \\
 \div 23 \\
 \hline
 2 \overline{) 57} \\
 \underline{46} \phantom{00} \\
 4 \overline{) 112} \\
 \underline{92} \phantom{00} \\
 9 \overline{) 208} \\
 \underline{207} \phantom{00} \\
 1
 \end{array}$$

#### 式十二

#### 式十三

我们为了去掉“减”过程的思路，可以令被除数变号，然后，整个“减”过程完全变成“加”过程。这可使整个运算的复杂性进一步降低。

以后，我们的除法就以此来进行。但，应该注意，此时若出现余数则要将该余数变号后，才是最终运算结果的余数。

4、《混十进制》{+\*} 与《普通十进制》{+} 的关系。

4.1 {+\*} 与 {+} 数的转换法

这里指整数的情况，例如 {+\*}  $3\bar{8}2\bar{2}\bar{9}6 = \{+\} 221716$  (式一)。

4.1.1 {+} 数本身即为 {+\*} 数的一种特况，故 {+} 数不经转换即为 {+\*} 数。

4.1.2 {+\*} 数转换成 {+}。方法有两种：一种将 {+\*} 数变为一正一负的两个 {+} 数求和。这有好多种。其中，典型的是将该 {+\*} 数中各正数字位及 0 位作为一正 {+} 数，而将各负数字位作为一负 {+} 数。

例 {+\*}  $3\bar{8}2\bar{2}\bar{9}6 = \{+\} 302006 - 20290 = 221716$

另一种方法是：{+\*} 数中，连续正数字（或 0）的数字段照写不变。如  $3 \times 2 \times \times 6$ 。但，当其不在 {+\*} 数末尾（个位）时，则最低位加  $\bar{1}$ ；连续负数字的数字段，则使负数字的相反正数字与所求转

换数字之和为 9, 如  $\times 1 \times 70 \times$ 。然后, 在其最低位加 1。

这样, 即得结果为 221716, 即为相应  $\{+\}$  数。

于是, 获得数  $\{+\}$  221716 即为所求结果。

(注: 式中负数字段右侧加上分段线。当不致误解时, 分段线可不划。)

#### 4.2 $\{+*\}$ 与 $\{+\}$ 对照表及其说明 (对照表见下面)

$0 = \bar{0} = 00 = 000 = \dots = \dot{0} = 0_+$	$\bar{0} = \bar{0}\bar{0} = \bar{0}\bar{0}\bar{0} = \dots = \dot{\bar{0}} = 0_-$
$1 = \bar{1} = 19 = 199 = \dots = 1\dot{9}$	$\bar{1} = \bar{1}9 = \bar{1}99 = \dots = \bar{1}\dot{9}$
$2 = \bar{2} = 18 = 198 = \dots = 1\dot{9}\bar{8}$	$\bar{2} = \bar{1}8 = \bar{1}98 = \dots = \bar{1}\dot{9}\bar{8}$
$3 = \bar{3} = 17 = 197 = \dots = 1\dot{9}\bar{7}$	$\bar{3} = \bar{1}7 = \bar{1}97 = \dots = \bar{1}\dot{9}\bar{7}$
$4 = \bar{4} = 16 = 196 = \dots = 1\dot{9}\bar{6}$	$\bar{4} = \bar{1}6 = \bar{1}96 = \dots = \bar{1}\dot{9}\bar{6}$
$5 = \bar{5} = 15 = 195 = \dots = 1\dot{9}\bar{5}$	$\bar{5} = \bar{1}5 = \bar{1}95 = \dots = \bar{1}\dot{9}\bar{5}$
$6 = \bar{6} = 14 = 194 = \dots = 1\dot{9}\bar{4}$	$\bar{6} = \bar{1}4 = \bar{1}94 = \dots = \bar{1}\dot{9}\bar{4}$
$7 = \bar{7} = 13 = 193 = \dots = 1\dot{9}\bar{3}$	$\bar{7} = \bar{1}3 = \bar{1}93 = \dots = \bar{1}\dot{9}\bar{3}$
$8 = \bar{8} = 12 = 192 = \dots = 1\dot{9}\bar{2}$	$\bar{8} = \bar{1}2 = \bar{1}92 = \dots = \bar{1}\dot{9}\bar{2}$
$9 = \bar{9} = 11 = 191 = \dots = 1\dot{9}\bar{1}$	$\bar{9} = \bar{1}1 = \bar{1}91 = \dots = \bar{1}\dot{9}\bar{1}$
$10 = \bar{10} = 10 = 190 = \dots = 1\dot{9}\bar{0}$	$\bar{10} = \begin{cases} \bar{1}0 = \bar{1}90 = \dots = \bar{1}\dot{9}\bar{0} \\ \bar{1}0 = \bar{1}90 = \dots = \bar{1}\dot{9}\bar{0} \end{cases}$
$11 = \bar{11} = 11 = 191 = \dots = 1\dot{9}\bar{1}$	$\bar{11} = \bar{1}1 = \bar{1}91 = \dots = \bar{1}\dot{9}\bar{1}$

说明: 式中  $\bar{9}$  表示为 9 的二次取负数 (二次以上从略), 余数同此。

① 式中  $0_+$   $0_-$  分别为从正负方向趋近于 0 所获得的 0;

② 式中  $\dot{9}$  表示连续任意非负整数位之一的 9, 读作“延 9”。式中  $\dot{0}$  表示连续任意非负整数位之一的 0, 读作“延 0”。这种数, 可以称为“无限延数”。

③ 无限延数有且仅有  $(\dot{0}, \dot{0}, \dot{9}, \dot{9})$  四种。由于  $\dot{0} = \dot{0}$ , 故无限延数有且仅有  $(\dot{9}, \dot{0}, \dot{9})$  三种。亦可写为  $(\dot{0}, \pm \dot{9})$ 。

④  $\bar{0} = 0$ , 由数 10 的两种表达形式可知。因此,  $\bar{0} = 0 = \dot{0} = \dot{0}$ 。

#### 4.3 $\{+*\}$ 与 $\{+\}$ 关系分析

4.3.1  $\{+\}$  数是  $\{+*\}$  数的一部分,  $\{+\}$  数集是  $\{+*\}$  数集的子集;

$\{+*\}$  数  $\supset \{+\}$  数, 即  $\{+*\}$  数对  $\{+\}$  数有包含关系。

4.3.2  $\{+\}$  数与  $\{+*\}$  数的关系是“一多对应”关系, 而不是“一一对应”关系。正由于此,  $\{+*\}$  就获得了多样处理的灵活性。这是  $\{+*\}$  运算中多样性、快速性的原因。从这一点来说,  $\{+*\}$

具有较强的功能。

4.3.3  $\{+*\}$  数转换为  $\{+\}$  数，只能化为相应唯一的一个数。这是因为， $\{+*\}$  数可经  $\{+\}$  数加减直接获得，而  $\{+\}$  数加减运算后的结果是唯一的。反之， $\{+\}$  数也只能化为相应唯一的一组  $\{+*\}$  无限延数。所以，这种  $\{+\}$  数的“一”与  $\{+*\}$  无限延数的“一”组两者是“一一对应”关系。

由此，可建立一种  $\{+*\}$  数与  $\{+\}$  数的互为映射关系。

由于变换是集到自身上的对应，所以：

$\{+\}$  与  $\{+*\}$  数是“一一变换”。对于运算系统来说， $\{+\}$  与  $\{+*\}$  数系统是“自同构”。相应  $\{+\}$  数的各种运算性质，亦在  $\{+*\}$  数系统中成立。

4.3.4  $\{+*\}$  中  $P>Q$ ，因而在该数制中自然数有时会出现多种形态表达，这正是该数制灵活性所在，它使运算得以简便快捷。也可以说  $\{+*\}$  是以多样性来换取了灵活性。

$\{+\}$  中  $P=Q$ ，因而在该数制中，自然数是连续唯一形态表达，它没有这种多样性。也缺少了这种相应的灵活性。

可以这么说，本发明的关键正是在此。有了它，才有了《混进方法 HJF》，才有了“笔算方式数字工程的”新技术方案。有了它，才有了电子计算机新技术方案。

4.3.5 应当指出，显然，上述对  $\{+\}$  及  $\{+*\}$  的分析，完全相应于  $\{Q\}$  及  $\{Q*\}$  的分析，因为  $\{+\}$  与  $\{Q\}$  是同构的。由此可知，①  $\{Q\}$  数与  $\{Q*\}$  数的关系是“一多对应”，而不是“一一对应”。②同时， $\{Q\}$  中的“一”个数与相应的  $\{Q*\}$  中的“一”组无限延数两者之间是“一一对应”关系。③  $\{Q\}$  与  $\{Q*\}$  数系统是“自同构”。相应  $\{Q\}$  数系统的各种运算性质，亦在  $\{Q*\}$  数系统中成立。

## 5、混 Q 进制 $\{Q*\}$ 及《混进方法 HJF》的应用

### 5.1 《混进方法 HJF》是一种优异的运算方法。

《混进方法 HJF》的理论和实践证明，它把混数与“进位行方法”紧密结合在一起，正好互补，互相促进，作用大大加强。于是  $+-\times\div$  四则运算（也就是有理数运算）全面、系统地改观。《混进方法 HJF》作为一种特别优异的运算方法，必将获得广泛的应用。

## 6、结论



综合上述，可有如下简明结论：

① 混 Q 进制  $\{Q^*\}$  及《混进方法 HJF》在有理数运算中可大大提高运算速度，而且大大降低笔算的出错率。

## 第二部分 混 Q 进制、进位行的处理器

四则运算是一切运算的基础，显然也是电子计算机的基础。

图 1 为本发明的混 Q 进制计算机总逻辑框图。由输入逻辑 101、CPU 中央处理器 102、外存 103、输出逻辑 104、控制台 105、输出转换逻辑 108 组成。而中央处理器 102 由内存 106、混 Q 运算控制逻辑 107 组成。这些部件的连接关系是本领域公知的。其中，普通 Q 进制数通过输入逻辑 101 输入中央处理器 102，并通过混 Q 运算控制逻辑 107 进行混 Q 运算，运算结果连接输出转换逻辑 108，结果以混 Q 进制数或者 Q 进制数通过输出逻辑 104 输出。内存 106 及外存 103 与运算控制逻辑 107 交换数据，执行原有普通 Q 进制的程序。总操作由控制台 105 控制按既定程序以时钟脉冲来实现。

图 2 为运算控制逻辑框图，由输入逻辑 101，k 重运算器 202，输出转换逻辑 108 及控制器 201 组成。

混 Q 进制数经移位寄存器输入逻辑 101 至 k 重运算器 202；k 重运算器 202 中，混 Q 进制数经 k 重运算获得混 Q 进制数的结果，经由输出转换逻辑（译码器）108 以 Q 进制数，或混 Q 进制数或普通十进制数通过输出逻辑 104 输出，控制器 204 协调控制整个运算控制器的逻辑。

图 3 为 K 重运算器的逻辑框图，由寄存器网 311；对冲网 312；划 Q 网 313 及累加器 304 组成。其中累加器（304）为每一位带有一个正负符号位的普通累加器，也可以为累加器的每一位从符号位寄存器分配一个符号位。寄存器网 311 由 A 寄存器 301、B 寄存器 302、……、2K 寄存器 303 组成。对冲网 312 由一个对冲逻辑 305 巡检；或由 K (2K-1) 个对冲逻辑 305、对冲逻辑 306、……、对冲逻辑 307 与寄存器网 311 中各个寄存器二二相连组成；或由分组、分级的对冲逻辑组成。划 Q 网 313 由一个划 Q 逻辑 310 巡检；或由 K (2K-1) 个划 Q 逻辑 308、划 Q 逻辑 309、…、划 Q 逻辑 310 与寄存器网 311 中各个寄存器二二相连组成；或由分组、分级的划 Q 逻辑组成。

寄存器网 311 及对冲网 312 及划 Q 网 313 组成“K 重运算器”。

上述“K重运算器”当K值较大时，可加以分级放大处理。

2K个寄存器中，前面K个寄存器存放输入的K个混Q数，每个寄存器以及累加器的每个位前附有符号位，该符号位为普通二态触发器。累加器仅为一个，它用来存放累加和数，累加器的每位前附有符号位，该符号位为普通二态触发器。符号位也可以放置在专用的符号位寄存器中，在运算时为存放混Q数的寄存器或累加器的每一位分配一个符号位。后面K个寄存器存放进位行数，组成k个进位行。

如果不采用对冲和划Q，则在运算过程中，其中两个寄存器的某位，经累加器累加得到该位的和与高位进位，其中进位送至任一进位行寄存器的相邻高位；当下一个运算指令到达时，将进位行与原存数送入累加器相加；

如此重复，最后再经累加器获得所求和数。

为了加快运算速度，可采用对冲网和划Q网，由控制器或程序发出的指令，先实施“对冲”及“划Q”运算，然后进行累加运算。

划Q产生的“进位”送至K重运算器中任一进位行寄存器的相邻高位的置“1”端。

图4为对冲逻辑（对冲器），由A寄存器的第i位401，B寄存器的第i位402，同逻辑403，异逻辑404及与门405组成。其中：A寄存器的第i位401，其前附有符号位，为普通二态触发器。其中 $A_i$ 的“1”端连接同逻辑403的输入、 $A_i$ 符的“1”端连接异逻辑404输入。

B寄存器的第i位402，其前附有符号位，为普通二态触发器。其 $B_i$ 的“1”端连接同逻辑403的输入， $B_i$ 符的“1”端连接异逻辑404的输入。同逻辑403的输出连接与门405输入；异逻辑404的输出连接与门405输入；与门405的输出连接A寄存器的第i位401的置“0”端及B寄存器第i位402的置“0”端。

图5为划Q逻辑（划Q器），由A寄存器的第i位501，B寄存器的第i位502，Q值判定逻辑503，同逻辑504及与门505组成，其中：A寄存器的第i位501，其前附有符号位，为普通二态触发器。其 $A_i$ 的“1”端连接Q值判定逻辑503的输入， $A_i$ 符的“1”端连接同逻辑504的输入。

B寄存器的第i位502，其前附有符号位，为普通二态触发器。其 $B_i$ 的“1”端连接Q值判定逻辑503的输入； $B_i$ 符的“1”端连接同逻辑

辑 504 的输入; Q 值判定逻辑 503 的输出连接与门 505 的输入; 同逻辑 504 的输出连接与门 505 输入; 与门 505 输出连接 A 寄存器的第 i 位 501 置“0”端及 B 寄存器第 i 位 502 的置“0”端。

当采用 {二\*} 运算时 (其他混数数制类似), 在运算及其控制中, 采用 {1̄, 0, 1} 三态进行。其中, 1̄、1 的正负号以一位 {二} 符号表示, 其权为 0。

当采用 {Q\*} 运算时, 运算器的输入不需要将 {Q} 数转换为 {Q\*} 数, 因为 {Q} 数本来就是 {Q\*} 数。即, {Q\*} 数 = {Q} 数 + 纯 {Q\*} 数。另一方面, 运算器的输出在一般中间过程也不必要将 {Q\*} 数转换为 {Q} 数。只有在需要输出最终结果时, 才将 {Q\*} 数转换为 {Q} 数 (实质是仅将纯 {Q\*} 数转换为 {Q} 数)。这时, 本发明的计算机在“运算”数字的输出界面上, 只需加上特别简单的 {Q\*} 转换到 {Q} 译码器即可。这一点在技术上不存在任何困难。原则上, 本发明的计算机其外存及输入输出端与现有 {Q} 电子计算机完全一样 (包括程序在内)。这其中的原因就在于, 全部 {Q} 数本身均为 {Q\*} 数所包含。在这种意义上, 现代 {Q} 数制电子计算机本来就是 {Q\*} 电子计算机的特况。

本发明的计算机系统中, 采用“多重运算器”。例如, 采用“八重运算器”。所谓“八重运算器”, 即将 8 个数放入 8 个寄存器中, 一次性完成加减运算。设多重数为 K,  $K=2^n \cdot 5^m$  较合适 (n、m 为非负整数)。主要是考虑 {二} 及 {十} 较普通常用。故 K=2、4、8、16、32、64、128……; 以及 K=10、20、40、80、160……; 以及 K=50、100、200……。其中, 较重要的可能是 K=8、10、16、20、32、40、50、64、80、100。同时, 乘法本质上原来就是连续加法, 除法本质上原来就是连续减法。因此, 在乘除中, 本发明的计算机亦可运用多重加减来处理。

本发明的计算机除采用一般的累加器运算外, 为了加速运算可以采用“对冲”及“划 Q”逻辑。所谓“对冲”, 即两相反数相加, 其和为零。所谓某位上的“划 Q”, 即 Q 进制的两数相加时, 其某位上按位加 ⊕ 和为零, 但产生进位 (其符号与两数符号一致)。“对冲”及“划 Q”逻辑线路在技术上是简单成熟的。见图 4 和图 5。

特别是, 在 {二\*} 电子计算机中, 仅仅只需先“对冲”后“划二”就能获得运算结果。当最终结果需要输出时, 才将 {二\*} 数转换成 {十} 数输出。

小结:

一、本发明的计算机是混  $Q$  进制  $\{Q^*\}$  的电子计算机, 是《混进方法 HJF》电子计算机。

二、混  $Q$  进制  $\{Q^*\}$  的电子计算机使现代以及未来基于其他原理上的各种电子计算机的运算速度大大提高。以八重运算器为例, 粗略地估算将使运算速度提高五倍。也就是说, 原 20 万次/s 的提高到 100 万次/s 左右; 原 20 亿次/s 的提高到 100 亿次/s 左右。

### 第三部分

#### 1. 增 $Q$ 进制 $\{Q^\Delta\}$ 及全一码

1.1 定义及符号 [文中凡未标明数制的数, 均指普通十进制数。下同。]

在一个数制中, 凡  $P=Q+1>Q$  的进制, 称为“增强  $Q$  进制”。简称为“增  $Q$  进制”, 以符号  $\{Q^\Delta\}$  来表示。显然,  $\{0, 1, 2\}$  二进制, 即为“增二进制  $\{2^\Delta\}$ ”;  $\{\bar{1}, 0, 1\}$  二进制也就是混二进制  $\{2^*$ , 亦为“增二进制  $\{2^\Delta\}$ ”。此外, 还有其他  $\{2^\Delta\}$ 。

#### 1.2 增一进制 $\{1^\Delta\}$ 及其运算

增  $Q$  进制  $\{Q^\Delta\}$  中, 当  $Q = 1$  时, 即为增一进制  $\{1^\Delta\}$ 。增一进制  $\{1^\Delta\}$  中, 主要有二种。其一是  $\{0, 1\}$  一进制, 其元器件为二态器件。它是人类历史上最早出现的位值制数制, 以实物的“有”与“无”两态来表示数; 其二是  $\{\bar{1}, 1\}$  一进制, 其元器件亦为二态器件, 它亦可表示全部整数。本文仅采用  $\{0, 1\}$  一进制来分析。

增一进制  $\{1^\Delta\}$  的运算。这里列出加法运算, 例如  $\{+\} 4 + 3 + 2 == 9 ==$

$\{1^\Delta\} 140101+4011+401=11001100040404011。$

#### 1.3 增一进制 $\{1^\Delta\}$ 与 $\{Q\}$ 的关系。

##### 1.3.1 $\{1^\Delta\}$ 数与 $\{Q\}$ 数的转换法。

$\{1^\Delta\}$  数转换成  $\{Q\}$  数, 可以将  $\{1^\Delta\}$  数中的各位数字 1, 以  $\{Q\}$  计数即可。所得  $\{Q\}$  计数和, 即为相应的  $\{Q\}$  数。这就是说,  $\{1^\Delta\}$  数中有几个 1, 则相应的  $\{Q\}$  数即为几。显然, 这是十分简单的法则。

$\{Q\}$  数转换成  $\{1^\Delta\}$  数, 可将  $\{Q\}$  数各位均乘以各位上的

权,然后将这些积以同样个数的 1, 分别在所要表达的  $\{-^{\Delta}\}$  数位置上, 以不重复的方式列出即可。这就是说,  $\{Q\}$  数为几, 则  $\{-^{\Delta}\}$  数中就有几个 1。显然, 这也是十分简单的法则。

### 1.3.2 $\{-^{\Delta}\}$ 数与 $\{Q\}$ 数对照表及其说明

见表一、三 (令  $Q=2、10$ )

见表二、四 (令  $Q=2、10$ )

$\{+\}$	$\{二\}$	$\{-^{\Delta}\}$	$\{-^{\Delta}\}$	$\{二\}$
			000	0
0	000	$0 \cdots 00000000 = \dot{0} = 0$	001	1
1	001	$0 \cdots 00000001 = 1 = 1\dot{0}$	010	1
2	010	$0 \cdots 00000011 = 11 = 11\dot{0} = 1\dot{0}1 = 1\dot{0}1\dot{0} = \cdots$	011	10
3	011	$0 \cdots 00000111 = 111 = 111\dot{0} = 11\dot{0}1 = 11\dot{0}1\dot{0} = \cdots$	100	1
4	100	$0 \cdots 00001111 = 1111 = 1111\dot{0} = 111\dot{0}1 = 111\dot{0}1\dot{0} = \cdots$	101	10
5	101	$0 \cdots 00011111 = 11111 = 11111\dot{0} = 1111\dot{0}1 = 1111\dot{0}1\dot{0} = \cdots$	110	10
6	110	$0 \cdots 00111111 = 111111 = 111111\dot{0} = 11111\dot{0}1 = 11111\dot{0}1\dot{0} = \cdots$	111	11
7	111	$0 \cdots 01111111 = 1111111 = 1111111\dot{0} = 111111\dot{0}1 = 111111\dot{0}1\dot{0} = \cdots$	= = = =	
=		=	1.2 表一	
		=	(Q = 2)	
		=	1.2 表二 (Q = 2)	
		=		

1					
1	1				
1	2	1			扬
1	3	3	1		辉
1	4	6	4	1	三
.	.				角
.	.				形

{+}	{0, 1}
0	$0 = \dot{0}$
1	$1 = 1 = 1\dot{0} = \dots$
2	$11 = 11\dot{0} = \dots 1\dot{0}1\dot{0} = \dots$
3	$111 = 111\dot{0} = \dots 11\dot{0}1\dot{0} = \dots$
4	=
5	
6	=
7	
8	=
9	$111111111 = 111111111\dot{0} = \dots 111111111\dot{0} = \dots$
10	$1111111111 = 1111111111\dot{0} = \dots = 1111111111\dot{0} = \dots$
=	=
=	=
=	=

表四

{-} = {0, 1} - {+}	
0000	0
0001	1
0010	1
0011	2
0100	1
0101	2
0110	2
0111	3
1000	1
1001	2
1010	2
1011	3
1100	2
1101	3
1110	3
1111	4

表三

说明：①  $\{-^\Delta\}$  数可表示全部  $\{Q\}$  数

②有较多的重复数，以 4 位  $\{-^\Delta\}$  数为例，除 0 及 4 唯一外，其余均有重复数。其中，1 有 4 个；2 有 6 个；3 有 4 个。于是，从 0~4 的重复数分别为 1, 4, 6, 4, 1 个。

这与二项式展开系数  $C_n^k$  是一致的。（位数  $n$  为自然数， $k$  为 0~ $n$ 。）

③表中 0 表示为任意非负整数位连续的 0。这与混  $Q$  进制中一样。称为“无限延数”。 $\{-^\Delta\}$  数中，无限延数有且仅有一个，即为“0”。

### 1.3.3 $\{-^\Delta\}$ 与 $\{Q\}$ 关系分析。

1.3.3.1  $Q \supset 1$ ,  $Q$  为自然数；1 为最小的自然数，也是最基础的自然数单元。 $Q$  包含 1，这使得相应的  $\{Q\}$  及  $\{-^\Delta\}$  之间存在自然的联系。

1.3.3.2  $\{Q\}$  数与  $\{-^\Delta\}$  数的关系是“一多对应”关系，而不是“一一对应”关系。正由于此， $\{-^\Delta\}$  就获得了多样处理的灵活性。这是  $\{-^\Delta\}$  运算中快速性的原因之一。从这一点来说， $\{-^\Delta\}$  具有较强的功能。

1.3.3.3  $\{-^\Delta\}$  数转换为  $\{Q\}$  数，只能化为相应唯一的一个数。这是因为， $\{-^\Delta\}$  数可经  $\{Q\}$  加减直接获得，而  $\{Q\}$  数加减运算

后的结果是唯一的。反之， $\{Q\}$ 也只能化为相应唯一的一组 $\{-^{\Delta}\}$ 无限延数。所以，这种 $\{Q\}$ 数的“一”与 $\{-^{\Delta}\}$ 无限延数的“一”组两者是“一一对应”关系。由此，可建立一种 $\{-^{\Delta}\}$ 数与 $\{Q\}$ 数的互为映射关系。对于运算系统来说， $\{Q\}$ 与 $\{-^{\Delta}\}$ 数系统是“同构”。相应 $\{Q\}$ 数的各种运算性质，亦在 $\{-^{\Delta}\}$ 数系统中成立。

1.3.3.4  $\{-^{\Delta}\}$ 中  $P=Q+1$   $Q$ ，因而在该数制中，自然数有时会出现多种形态表达，这正是该数制灵活性所在，它使得运算得以简便快捷。也可以说， $\{-^{\Delta}\}$ 是以多样性来换取了灵活性。

$\{Q\}$ 中  $P=Q$ ，因而在该类数中，自然数是连续唯一形态表达。它没有这种多样性，也缺少了这种相应的灵活性。

1.3.3.5 上述 $\{-^{\Delta}\}$ 与 $\{Q^*\}$ 相结合，使得功能更加增强。考虑到 $\{-^{\Delta}\} \rightarrow \{Q\} \rightarrow \{Q^*\}$ 这其中有着内在的联系，显然，这一切均在预料之中。

#### 1.4 增一进制 $\{-^{\Delta}\}$ 的应用

1.4.1 增一进制 $\{-^{\Delta}\}$ 的运算是一种优异的运算。由于它以权为1的单元1配以0构造数，故其运算中常以“传送”来实现。至于 $\{-^{\Delta}\}$ 数运算中的“进位”也可以当前位的二数按位加和为0，而进位为 $Q$ 的“划 $Q$ ”逻辑实现。这种“传送”及“划 $Q$ ”的逻辑实现，结构特别简单，速度却特别的快。这是 $\{-^{\Delta}\}$ 数运算中快速性原因之二。

当 $\{-^{\Delta}\}$ 数与纯 $\{Q^*\}$ 数结合运算时，又补充了“对冲”这一结构更为简单、速度更为快速的逻辑。这是 $\{-^{\Delta}\}$ 数运算中快速性原因之三。

1.4.2  $\{-^{\Delta}\}$ 与 $\{Q^*\}$ 结合可作为多种新一代超高速电子计算机的技术方案。[详见下一节。]

### 2. 全一进制、全一数及全一码

#### 2.1 全一进制和全一数

增一进制 $\{-^{\Delta}\}$ 数的多样性是 $\{-^{\Delta}\}$ 数运算快速的原因之一。 $\{-^{\Delta}\}$ 数在“多重运算”时，在没有必要获得最终结果的过程运算中，产生的每一重数据均保留在相应的多重寄存器中作为中间结果。

但是，由于 $\{-^{\Delta}\}$ 数具有极端的多样，常造成数运算形式难以把握。因此，在一般情况下，有必要对 $\{-^{\Delta}\}$ 数加以某种约束条件，

使其减小多样性。这就产生了“全一进制”。

在增一进制  $\{-^{\Delta}\}$  的正整数中，限定每一组无限延数，只选取从个位开始，从右向左连续排列 1 的唯一的一种形态表达。例如： $\{+\}$  数  $3 = \{-^{\Delta}\}$  数  $111/111\dot{0}/11\dot{0}1/\dots$ （/表“或者”），限定为  $\{+\}$   $3 = \{-^{\Delta}\}$   $111$ 。这样，每一组无限延数中的重复数均被删除，只剩下一个全是 1 的唯一形态。我们称为“全一数”。表达“全一数”的进制称之为“全一进制”。表二、四中， $\{-^{\Delta}\}$  数最左边的形态，即为“全一进制”数。

因此，“全一进制”是加特定约束条件的增一进制  $\{-^{\Delta}\}$ 。

## 2.2 全一码

全一进制显然具有如下优缺点。优点：①运算速度快。“传送”代替了“翻转”。②多重运算时，不需要二、二求和，只需要先“对冲”及后“划 Q”即可得结果。这就大大加快了总体运算速度。③与  $\{Q\}$  转换方便。缺点：①“字长”太长，位数多。②荷载信息量较小。因此，根据全一进制的优缺点，扬长避短，以全一进制来编码  $\{Q\}$  是合适的。以“全一进制”来编码，称为“全一编码”。“全一编码”中采用的“全一数”，称为“全一码”。表一，显示出全一码一位，编码  $\{二\}$  数元的情况。由表一可见，全一码一位编码的  $\{二\}$  数，即为  $\{二\}$  数本身。表二，显示出以全一码九位，编码  $\{十\}$  数元的情况。由表二可见，全一码九位编码的  $\{十\}$ ，字长增加至 9 倍。

全一码	$\{二\}$ 数元	全一码	$\{十\}$
0	0	0	0
1	1	1	}
		11	
		111111111	
2.2 表一		===	
		111111111	9

2.2 表二

例如： $\{十\} 23 = \text{全一码} = \equiv$ 。

对于混 Q 进制  $\{Q^*\}$ ，则可以全一码加符号位来编码。特别是，这里全一码一位编码的  $\{二^*\}$  数，即为  $\{二^*\}$  数本身；这里  $\{十^*\}$  数，则为全一码加符号位来编码。



### 2.3 全一码的计算。

全一码的计算非常简单。以二数加法为例，仅为二数中 1 的不重复排列，简称为“排 1”。如  $11+111=11111$ 。

### 2.4 全一码的应用。

全一码主要应用于对  $\{Q\}$  及  $\{Q^*\}$  数进行编码。特别是，

①采用全一码九位编码  $\{十\}$  数可以实现本发明的普通十进制  $\{十\}$ 、全一码电子计算机。

②采用全一码九位编码  $\{十^*\}$  数可以实现本发明的混十进制  $\{十^*\}$ 、全一码电子计算机。

③采用全一码编码  $\{Q^*\}$  数可以实现本发明的混  $Q$  进制  $\{Q^*\}$ 、进位行、全一码、电子计算机。

## 第四部分 混二进制 $\{二^*\}$ 、进位行处理器技术方案

该本发明的计算机是在  $\{二\}$  数制电子计算机基础上，将原来采用的  $\{二\}$  数制改变成包含它本身在内的  $\{二^*\}$  数制。它可以视为全一码一位加符号位来编码的  $\{二^*\}$  电子计算机，简称为混二进制  $\{二^*\}$  电子计算机。该电子计算机总逻辑框图如图 1 所示。

如现有的电子计算机为  $\{十\}$  数制的，则将原来所采用的  $\{十\}$  数制改变成包含它本身在内的  $\{十^*\}$  数制。

如现有的电子计算机为  $\{Q\}$  数制的，则将所采用的  $\{Q\}$  数制改变成包含它本身在内的  $\{Q^*\}$  数制。

当具备三态存储器或在存储量较小的专用计算机中，可以设计本发明的计算机采用数制为  $\{Q^*\}$ ，特别是  $\{二^*\}$ ；也可能采用另一类混数数制，即“含 0、整数段、对称”数元集的  $\{\bar{1}, 0, 1\}$  三进制等奇数普通数制。

本发明的计算机的运算采用《混进方法 HJF》。即，混二进制  $\{二^*\}$  的《混进方法 HJF》；或混十进制  $\{十^*\}$  的《混进方法 HJF》；或者，其他混  $Q$  进制  $\{Q^*\}$  的《混进方法 HJF》。

另一方面，亦可采用  $\{\bar{1}, 0, 1\}$  三进制的《混进方法 HJF》；或者，其他“含 0、整数段、对称”数元集的奇数普通数制的《混进方法 HJF》。

## 第五部分 普通十进制 $\{十\}$ 、全一码电子计算机新一代技术方

## 案

(一) 该普通十进制 {十}、全一码电子计算机是在 {十} 电子计算机基础上, 以全一码九位来编码的 {十} 电子计算机。

(二) 该电子计算机总逻辑框图, 如图 1 所示。

当采用普通十进制 {十}、全一码运算时, 由于该电子计算机本身就是 {十} 计算机, 运算器的输入及输出均通过极其简单的全一码译码器来与 {十} 数相互转换。这就避免了二进制 {二} 计算机中, 需要采用 8421 编码等来与十进制 {十} 数相互转换的问题。人类历史中, {十} 计算应用的广度和深度, 均是其他进制所不能比拟的。人类长期历史文化的底蕴与沉积, 使得 {十} 具有牢不可破的至尊地位。因此, {十} 全一码电子计算机具有特别重要的意义。

在图 1 的输出转换逻辑中, 全一码九位编码的 {十} 数, 通过全一码译码器变换成形式规范的 {十} 数。{十} 数输入时, {十} 数以全一码来编码。这一点在计算机技术中是成熟的。

(三) 普通十进制 {十}、全一码电子计算机采用“划十”逻辑, 即可获得运算结果。所谓某位上“划十”, 即十进制的两数相加时, 其某位上按位和 $\oplus$ 为零, 但产生进位。“划十”逻辑线路在技术上也是简单的、成熟的。

小结: 一、普通十进制 {十}、全一码电子计算机一般可作为专用机。

## 第六部分 混十进制 {十\*}、全一码电子计算机新一代技术方案

(一) 该混十进制 {十\*}、全一码电子计算机是在 {十} 电子计算机基础上, 将原来采用的 {十} 数制改变成包含它本身在内的 {十\*} 数制。

(二) 混十进制 {十\*}、全一码电子计算机, 是全一码九位加符号位来编码的 {十\*} 电子计算机。

(三) 该电子计算机的运算采用《混进方法 HJF》。即, 混十进制 {十\*} 的《混进方法 HJF》。

(四) 混十进制 {十\*}、全一码电子计算机总逻辑框图如说明

书附图 1 所示。

混十进制  $\{十^*\}$ 、全一码电子计算机中， $\{十\}$  数以全一码九位来编码； $\{十^*\}$  数以全一码九位加一位  $\{0, 1\}$  符号位来编码输入及输出。

当采用  $\{十^*\}$  运算时，运算器的输入不需要将  $\{十\}$  数转换为  $\{十^*\}$  数，因为  $\{十\}$  数本来就是  $\{十^*\}$  数。即， $\{十^*\}$  数 =  $\{十\}$  数 + 纯  $\{十^*\}$  数。另一方面，在一般中间过程，运算器的输出，也不必要将  $\{十^*\}$  数转换为  $\{十\}$  数。只有在需要输出最终结果时，才将  $\{十^*\}$  数转换为  $\{十\}$  数（实质是仅将纯  $\{十^*\}$  数转换为  $\{十\}$  数）。这时，本发明的计算机在“运算”的输出界面上，只需加上特别简单的全一码编码的  $\{十^*\}$  转换到  $\{十\}$  的译码器即可。这一点在技术上不存在任何困难。原则上，本发明的计算机其内外存及输入输出端与现有  $\{十\}$  电子计算机完全一样（包括程序在内）。这其中的原因就在于，全部  $\{十\}$  数本身均为  $\{十^*\}$  数所包含。在这种意义上，现代  $\{十\}$  数制电子计算机本来就是  $\{十^*\}$  电子计算机的特况。

（五）混十进制  $\{十^*\}$ 、全一码电子计算机系统中，采用“多重运算器”。例如，采用“八重运算器”。所谓“八重运算器”，即将 8 个数放入 8 个寄存器中，一次性完成加减运算。设多重数为  $K$ ， $K=2^n \cdot 5^m$  较合适（ $n, m$  为非负整数）。主要是考虑  $\{二\}$  及  $\{十\}$  较普通常用。故  $K=2, 4, 8, 16, 32, 64, 128, \dots$ ；以及  $K=10, 20, 40, 80, 160, \dots$ ；以及  $K=50, 100, 200, \dots$ 。其中，较重要的可能是  $K=8, 10, 16, 20, 32, 40, 50, 64, 80, 100$ 。同时，乘法本质上原来就是连续加法，除法本质上原来就是连续减法。因此，在乘除中，本发明的计算机亦可运用多重加减来处理。

（六）混十进制  $\{十^*\}$ 、全一码电子计算机采用“对冲”及“划十”逻辑。所谓“对冲”，即两相反数相加，其和为零。所谓某位上的“划十”，即十进制的两数相加时，其某位上按位加  $\oplus$  和为零，但产生进位（其符号与两数符号一致）。“对冲”及“划十”逻辑线路在技术上是简单成熟的。见图 4 和图 5。

特别是，在  $\{十^*\}$  全一码电子计算机中，仅仅只需先“对冲”、后“划十”就能获得运算结果。当最终结果需要输出时，才将全一码编码的  $\{十^*\}$  数转换成  $\{十\}$  数输出。

小结:

一、{十\*}全一码电子计算机是混十进制{十\*}的电子计算机，是《混进方法 HJF》电子计算机。

二、本发明的{十\*}全一码电子计算机使现代以及未来基于其他原理上的各种电子计算机运算速度大大提高。以八重运算器为例，粗略地估算将使运算速度提高五倍以上。也就是说，原 20 万次/s 的提高到 100 万次/s 左右；原 20 亿次/s 的提高到 100 亿次/s 左右。

## 权 利 要 求

1. 一种混 Q 进制、进位行数字工程方法，包括以下步骤：

第 1 步，将参与运算的普通 Q 进制数的每一位数字都加上一个数符，即表示该位数为正或负，使它成为混 Q 进制数，设，参予运算的数为 k 个混 Q 进制数，

第 2 步，对 k 个数同时进行混 Q 进制的求和运算，从最低位开始，按位相加，即在某一位上，取前述 k 个数中的二个数按位相加，得到“按位和”，即为该位这二个数相加的和数，将此和数记入下一运算层，作为“部份和”数，同时所得“混数进位”，则存放到下一运算层中的任一进位行中与该位相邻的高位处；

第 3 步，在该位上取 k 个数中的另二个数，进行第二步的运算，如此反复，直至 k 个数均取完为止；当 k 个数中仅剩下一个数时，则直接移至下一运算层的同一位上作为“部份和”数；

第 4 步，在上述某位的相邻高位上，重复第 2 步及第 3 步的运算，直至 k 个运算数的每一位都全部操作结束；

第 5 步，在下一个运算层中，将上述“按位和”数与进位行中的“进位数”进行前述第 2 步、第 3 步、第 4 步求和运算；

重复第 2 步至第 5 步的运算，直至不产生“混 Q 进位”为止，则最后一次“按位加”所得和数，即为所求加法运算结果。

2. 如权利要求 1 的混 Q 进制、进位行数字工程方法，其特征在于在某一位上，对 k 个数中的二个数进行求和运算时，如果其中两个运算数的该位为相反数，则该位和为零，然后将该两个运算数的某位均以逻辑方式置“0”，不再参加以后的运算；如果在某一位上，对 k 个数中的二个数进行求和运算时，其中两个运算数的按位加和为零，但产生进位，则将其进位放入任一进位行中的相邻高位，然后将该两个运算数的某位均以逻辑方式置“0”，不再参加以后的运算。

3. 如权利要求 1 或 2 的混 Q 进制、进位行数字工程方法，其特征在于以全一码来编码混 Q 进制数，即将各个混 Q 进制数的每一位数 S，都以 S 个 1 从最低位顺序至高位排列来对应，其余高位均为 0，总位数则为 (Q-1) 位；同时，将该位的数符，即表示该位的数为正或负，作

为相应全一码中每一位上的数符。

4. 如权利要求 1-3 任一个的混  $Q$  进制、进位行数字工程方法，其特征在于二数加法仅为二数中 1 的不重复排列。

5. 权利要求 1 或 2 的混  $Q$  进制、进位行数字工程方法，其中所述运算数是混  $Q$  进制数， $Q$  为自然数。

6. 权利要求 1 或 2 的混  $Q$  进制、进位行数字工程方法，其中所述运算数是普通混  $Q$  进制数，特别是普通混三进制数。

7. 权利要求 1 或 2 的混  $Q$  进制、进位行数字工程方法，其中所述运算数是混数数制数。

8. 一种混  $Q$  进制、进位行处理器，包括：输入逻辑 (101)， $k$  重运算器 (202)，输出转换逻辑 (108) 及控制器 (201) 组成；混  $Q$  进制数移位寄存器输入逻辑 (101) 至  $k$  重运算器 (202)； $k$  重运算器 (202) 中，混  $Q$  进制数经  $k$  重运算获得混  $Q$  进制数的结果，经由译码器输出转换逻辑 (108) 以  $Q$  进制数、或混  $Q$  进制数、或普通十进制数通过输出逻辑 (104) 输出，控制器 (201) 协调控制整个运算控制器的逻辑；其中，

为  $2K$  个寄存器中的每个寄存器以及累加器的每一位分配一个符号位，该符号位为普通二态触发器；前面  $K$  个寄存器存放输入的  $K$  个混  $Q$  数，后面  $K$  个寄存器组成  $K$  个进位行；

在运算过程中，其中两个寄存器的某位，经累加器累加得到该位的和与高位进位，其中进位送至任一进位行寄存器的相邻高位；当下一个运算指令到达时，将进位行与原存数送入累加器相加；

如此重复，最后再经累加器获得所求和数。

9. 如权利要求 8 的混  $Q$  进制、进位行处理器，进一步包含：

对冲网 (312) 和划  $Q$  网 (313) 与寄存器网 (311) 中各个寄存器二二相连组成；

由控制器或程序发出的指令，先对某位上运算数的各个数实施“对冲”及“划  $Q$ ”运算，然后进行累加运算；其中累加器 (304) 为每一位带有一个正负符号位的普通累加器；

某位上划  $Q$  产生的“进位”，送至  $K$  重运算器 (202) 中任一进位行寄存器的相邻高位置“1”端。

10. 如权利要求 9 的混  $Q$  进制、进位行处理器，其中的对冲网 (312)

由一个对冲逻辑(305)巡检,或由  $K(2K-1)$  个对冲逻辑(305、306、...、307)与寄存器网(311)中各个寄存器二二相连组成;或由分组、分级的对冲逻辑组成;

其中的划Q网(313)由一个划Q逻辑(310)巡检,或由  $K(2K-1)$  个划Q逻辑(308、309、...、310)与寄存器网(311)中各个寄存器二二相连组成或由分组、分级的划Q逻辑组成;

上述“K重运算器”当K值较大时,可以进行分级放大。

11. 如权利要求10的混Q进制、进位行处理器,其中的对冲逻辑由A寄存器的第i位(401),B寄存器的第i位(402),同逻辑(403),异逻辑(404)及与门(405)组成,其中:A寄存器的第i位(401),其前附有符号位,为普通二态触发器;其  $A_i$  的“1”端连接同逻辑(403)的输入、 $A_i$  符的“1”端连接异逻辑(404)输入;B寄存器的第i位(402),其前附有符号位,为普通二态触发器;其  $B_i$  的“1”端连接同逻辑(403)的输入, $B_i$  符的“1”端连接异逻辑(404)的输入;同逻辑(403)的输出连接与门(405)输入;异逻辑(404)的输出连接与门(405)输入;与门(405)的输出连接A寄存器的第i位(401)的置“0”端及B寄存器第i位(402)的置“0”端;

其中的划Q逻辑由A寄存器的第i位(501),B寄存器的第i位(502),Q值判定逻辑(503),同逻辑(504)及与门(505)组成,其中:A寄存器的第i位(501),其前附有符号位,为普通二态触发器;其  $A_i$  的“1”端连接Q值判定逻辑(503)的输入, $A_i$  符的“1”端连接同逻辑(504)的输入;B寄存器的第i位(502),其前附有符号位,为普通二态触发器;其  $B_i$  的“1”端连接Q值判定逻辑(503)的输入; $B_i$  符的“1”端连接同逻辑(504)的输入;Q值判定逻辑(503)的输出连接与门(505)的输入;同逻辑504的输出连接与门(505)输入;与门(505)输出连接A寄存器的第i位(501)置“0”端及B寄存器第i位(502)的置“0”端。

12. 如权利要求8的混Q进制、进位行处理器,其中所述运算数用全一码编码来表示。

13. 如权利要求8的混Q进制、进位行处理器,其中所述运算数是混Q进制数,Q为自然数。

14. 如权利要求8的混Q进制、进位行处理器,其中所述运算数是

普通 Q 进制数。

15. 如权利要求 8 的混 Q 进制、进位行处理器，其中所述运算数是混数数制数。



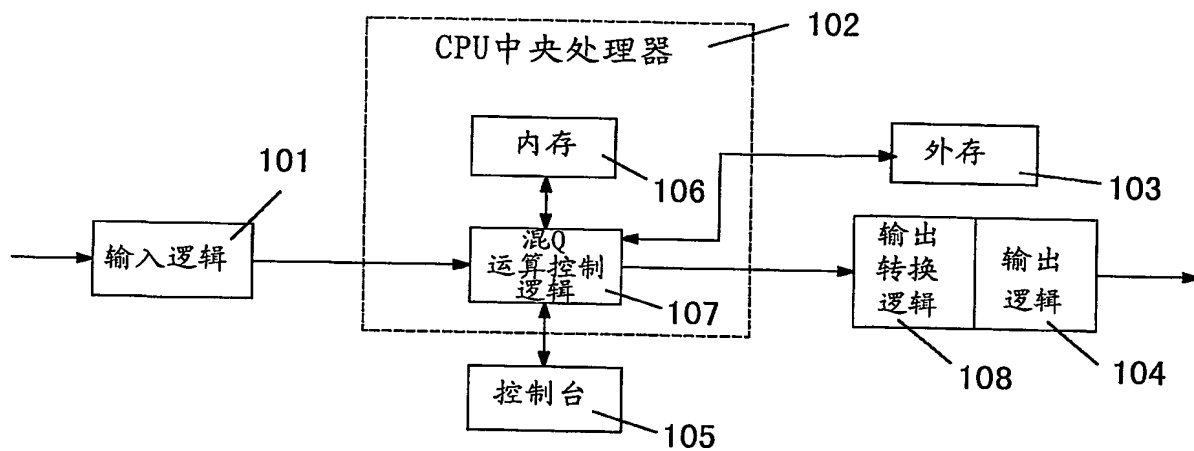


图 1

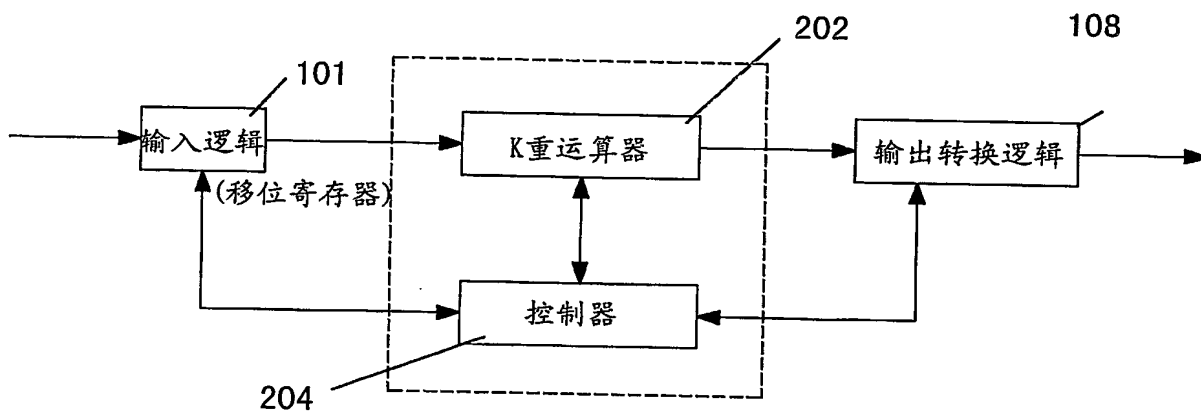


图 2

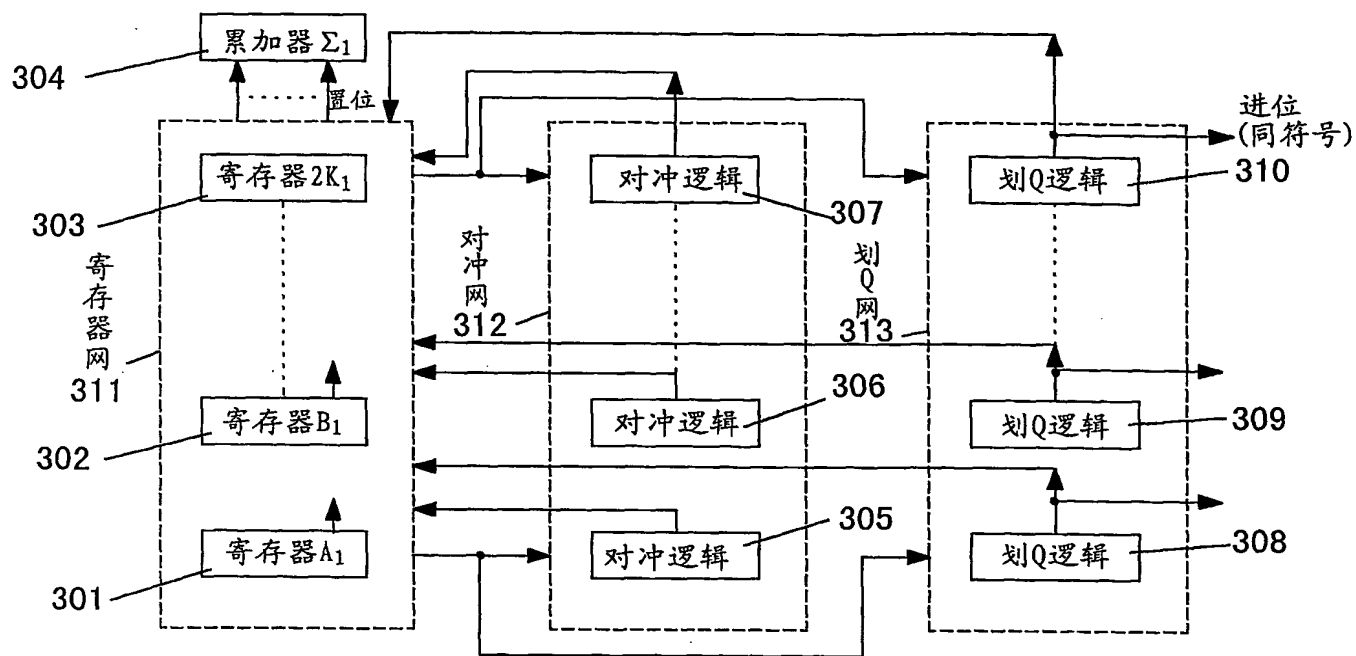


图 3

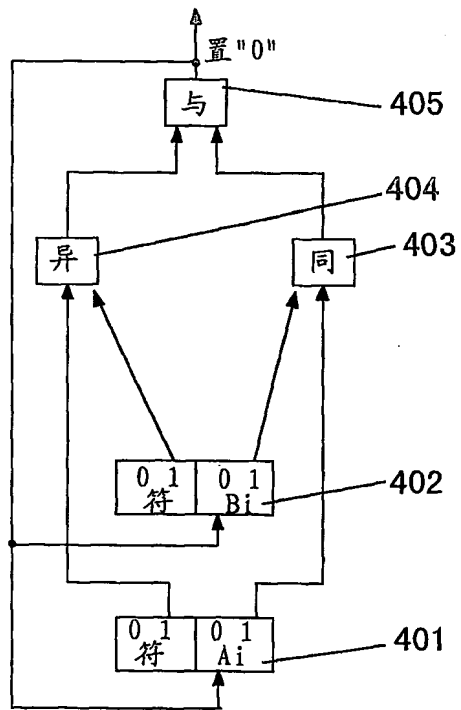


图 4

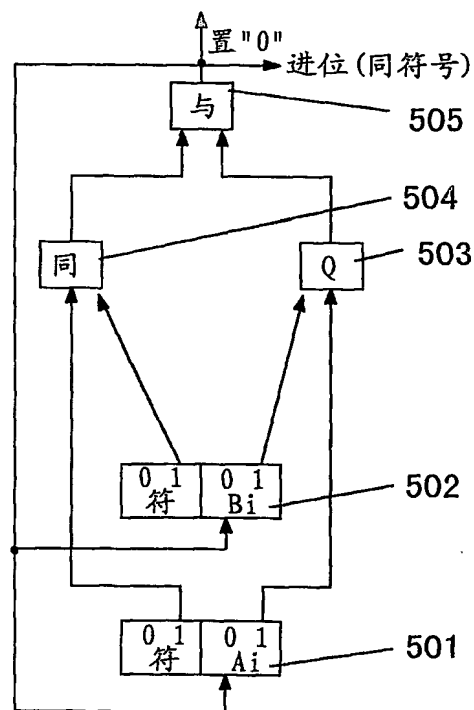


图 5

# INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2004/000375

## A. CLASSIFICATION OF SUBJECT MATTER

IPC7 G06F7/00

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC7 G06F7/00, 9/00

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNPAT,WPL,EPODOC,PAJ

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US6073149A (Yozan,INC.,Tokyo,Japan) 06.06.2000 see whole document	1-15
A	CN1136680A (Samsung ELECTRONICS.CO.LTD) 27.11.1996 see whole document	1-15

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

\* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"I" document which may throw doubts on priority claim (S) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

9 Jul 2004(9.7.2004)

Date of mailing of the international search report

22 · JUL 2004 (22 · 07 · 2004)

Name and mailing address of the ISA/

6 Xitucheng RD., Jimen Bridge, Haidian District,  
100088 Beijing, China

Facsimile No. (86-10)62019451

Authorized officer



Telephone No. (86-10)62084990

**INTERNATIONAL SEARCH REPORT**  
Information on patent family members

International application No.  
PCT/CN2004/000375

US6073149A	06.06.2000	EP0872794A2	21.10.1998
		CN1197952A	04.11.1998
		JP11003389A	06.01.1999
		JP10289285A	27.10.1998
CN1136680A	27.11.1996	DE19545900A	28.11.1996
		KR158647B	15.12.1998
		JP8314697A	29.11.1996
		US5870322A	09.02.1999

## 国际检索报告

国际申请号

PCT/CN2004/000375

## A. 主题的分类

IPC7 G06F 7/00

按照国际专利分类表(IPC)或者同时按照国家分类和 IPC 两种分类

## B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

IPC7 G06F 7/00, 9/00

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称, 和使用的检索词(如使用))

CNPAT, WPI, EPODOC, PAJ: 运算, 进位, 符号, 中间和, 加法器, 运算器, 处理器; operation, carry, symbol+, signal, intermediate sum, addition, ALU, processor

## C. 相关文件

类 型*	引用文件, 必要时, 指明相关段落	相关的权利要求
A	US6073149A (株式会社鹰山) (06.06 月 2000) 参见全文	1-15
A	CN1136680A (三星电子株式会社) (27.11 月 1996) 参见全文	1-15

☐ 其余文件在 C 栏的续页中列出。☒ 见同族专利附件。

\* 引用文件的具体类型:

“A” 认为不特别相关的表示了现有技术一般状态的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件, 为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布, 与申请不相抵触, 但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件, 单独考虑该文件, 认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件, 当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时, 要求保护的发明不具有创造性

“&amp;” 同族专利的文件

国际检索实际完成的日期

09. 07 月 2004 (09.07.2004)

国际检索报告邮寄日期

22. 7 月 2004 (22. 07. 2004)

中华人民共和国国家知识产权局(ISA/CN)

中国北京市海淀区蓟门桥西土城路 6 号 100088

传真号: (86-10)62019451

受权官员

A413



电话号码: (86-10)62084990

国际检索报告  
关于同族专利的信息

国际申请号  
PCT/CN2004/000375

检索报告中引用的 专利文件	公布日期	同族专利	公布日期
US6073149A	06.06.2000	EP0872794A2	21.10.1998
		CN1197952A	04.11.1998
		JP11003389A	06.01.1999
		JP10289285A	27.10.1998
CN1136680A	27.11.1996	DE19545900A	28.11.1996
		KR158647B	15.12.1998
		JP8314697A	29.11.1996
		US5870322A	09.02.1999